
Navigation-based Constrained Trajectory Generation for Advanced Driver Assistance Systems

**J  r  mie Daniel, Abderazik Birouche,
Jean-Philippe Lauffenburger and Michel
Basset**

Universit   de Haute Alsace (UHA) - MIPS laboratory
12 rue des Fr  res Lumi  re 68093 Mulhouse Cedex, France.
E-mail: jeremie.daniel@uha.fr
E-mail: abderazik.birouche@uha.fr
E-mail: jean-philippe.lauffenburger@uha.fr
E-mail: michel.basset@uha.fr.

Abstract: This paper presents a new approach to constrained trajectory generation dedicated to Advanced Driver Assistance Systems. Based on the information provided by the digital map database of a navigation system, the proposed solution is devoted to a control-oriented trajectory generation approach taking account of constraints which limit the behaviour of a common car, relative to the road to be followed and finally linked to the driver. Characteristically, these trajectories stay within the traffic lane borders; at the same time, they minimize the energy along the path, and finally, they are curvature continuous. The present trajectory generation has been tested on a specific test track and the results show the efficiency of the proposed solution.

Keywords: ADAS, digital map database, constrained trajectory generation, cubic spline, convex optimization.

Reference to this paper should be made as follows: Daniel, J. and Birouche, A. and Lauffenburger, J.P. and Basset, M. (20 xx) ‘Navigation-based Constrained Trajectory Generation for Advanced Driver Assistance Systems’, *Int. J. Vehicle Autonomous Systems*, Vol. x, No. x, pp.xxx-xxx.

Biographical notes: Jeremie Daniel obtained a BSc and a MSc in electrical engineering and computer science at UHA in 2005 and 2007 respectively. He is currently preparing for a PhD degree in electronics, automation and computer engineering in the Modelling Process Intelligence Systems (MIPS) laboratory. His research interests include data fusion and path planning for navigation-based Advanced Driver Assistance Systems.

Abderazik Birouche obtained a MSc and a PhD in automatic control at the Institut National Polytechnique de Lorraine (INPL), France in 2003 and 2006 respectively. Since 2007, Abderazik Birouche has been a

research engineer at the MIPS laboratory. His research interests include hybrid systems, trajectory planning with applications in the vehicle domain.

Jean-Philippe Lauffenburger obtained a MSc and a PhD in electrical engineering at UHA respectively in 1998 and 2002. Since 2004, he has been a lecturer at the UHA. His major research interests are data fusion applied to the design of advanced control applications, and to objective/subjective correlation. He is an active member of the IFAC “Transportation Systems” Technical Committee 7.4.

Michel Basset has been Professor in control engineering at UHA since 2005. He obtained a PhD in automatic control (1991) at UHA. In 1992, he joined the MIPS laboratory to continue modeling and control activities with applications in the automotive domain. He became lecturer in the same time. In 2003, he was appointed Senior lecturer and later Professor. He is an active member of the IFAC “Automotive Control” Technical Committee 7.1.

1 Introduction

The navigation domain has become quite popular in the last decade. For drivers, it is now as natural to know their position on a map as to know the current time. It gives the driver and passengers a feeling of greater comfort, as it prevents them from losing their way.

Navigation systems also play an important role in Advanced Driver Assistance Systems (ADAS). Indeed, navigation systems act as virtual sensors providing information about both the current and upcoming road context contrary to other sensors which only give information about the current road context. However, navigation systems are not exempt of limitations: according to several projects such as ACTMAP (Thomas et al. (2008)) or EDMAP (CAMP (2004)), current navigation systems do not provide sufficient information for control-oriented ADAS. This is due to the average reality approximation of digital map databases. Indeed, for memory and cost limitations, navigation devices do not contain all road context information. One of the consequences is that roads are vectorized, in other words, represented by a succession of road centreline points.

A solution to overcome these digital map database limitations and inaccuracies for control-oriented ADAS lies in the computation of trajectories. Indeed, this solution, which has been widely used in different domains, can provide a set of continuous information which is helpful for control-oriented applications. In addition, most of the path generation solutions include constraints which help to define the optimal or suboptimal trajectory, for example by taking a robot’s geometric or dynamic limitations into account (Gomez et al. (2008)), or by predicting a robot’s motion (Beji and Abichou (2009)). There are numerous trajectory generation solutions, based on the use of several mathematical models. Parametric Cubic Splines (De Boor (1978)) are used here, because they represent a

straightforward interpolation, they have a curvature continuity property, and they are adapted to all road contexts (bend, straight line, roundabout, etc.).

This paper presents a new constrained trajectory generation approach to control-oriented ADAS, based on the information provided by a digital map database. Considering the digital map limitations (CAMP (2004)), the proposed solution is a map-based continuous trajectory generation formulated as an optimization problem. Contrary to recent studies such as Li et al. (2008) or Eidehall et al. (2007) which focus on road curve reconstruction and curvature estimation, the present study is dedicated to the definition of a curvature-continuous trajectory for ADAS control applications, such as Curve Speed Assistance (Li et al. (2008)), Longitudinal Control (Daniel et al. (2009)), etc.

The main contribution of this paper lies in the integration and in the generation process of different constraints related to the road, to the vehicle and to the driver, while considering the digital map database limitations. This constrained generation is formulated as a convex optimization problem which tends to minimize a given cost criterion. If most optimizations minimize the time required to cover the trajectory or the trajectory distance, here the strain energy of the trajectory is minimized. As this criterion is directly linked to the curvature, its minimization leads to the generation of smoother trajectories. To check the validity of the present solution, tests were carried out using real road data provided by a navigation system. The results have shown that this new method leads to the generation of safe, smooth curvature continuous lane trajectories.

After the presentation of the work context in Section 2, Section 3 depicts the considered constraints and the mathematical model used. This is followed by the constrained trajectory generation proposal in Section 4. The results are then presented in Section 5, followed by a conclusion.

2 Work Context

2.1 Digital Map Database overview

Nowadays, digital map databases contain numerous geographical, topological, and informative attributes of the real driving infrastructure such as traffic signs, intersection locations, number of driving lanes, etc. These attributes can have different levels of abstraction. For example, the number of lanes is related to a road portion while the presence of road signs is related to specific road points. Fig.1 presents a schematic but non exhaustive overview of current digital map database composition. The road network vectorization, based on a succession of specific points, is clearly shown. It indeed divides road into vectors, also called segments, linked by nodes which usually correspond to intersections. Segments can also contain shape points which give more precise information about the current road geometry, specific attributes, etc. This figure also describes the Electronic Horizon (EH) concept. An *EH* (light grey lines in Fig.1) is a road network containing the set of possible roads (taking account of the driving rules) which the vehicle is likely to use in the close future. It can be interpreted as a selection of the relevant information in the large amount of data provided by the *Digital Map Database*. This process is carried out by a so-called Electronic Horizon Provider (EHP) which

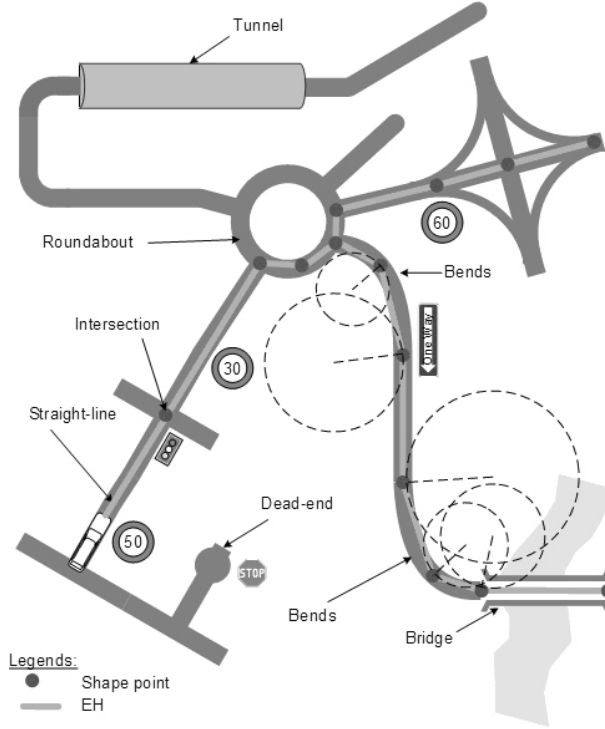


Figure 1 Digital Map Database Contents

processes the map data in order to provide the Electronic Horizon. The latter is generated relative to the vehicle position projected on the map using a vehicle-to-map correlation technique (map-matching).

As roads are approximated by a succession of points, the road context information is mainly related to these points. Consequently, the digital map database reality approximation involves discretized and discontinuous information. Moreover, it is important to note that shape points and nodes always refer to the middle of the road; there is also no information about the current driving lane. This may be problematic for control systems which require, for instance, continuous in-lane curvature. Finally, note that digital map data are subject to biases w.r.t real network. For current navigation systems an absolute error up to $5m$ and a relative error up to $2m$ are common. These errors correspond to the difference between the digitalized points and their real position on the road (cf. Fig.2 grey circles). Due to these errors, it is obvious that conventional interpolation approaches, directly based on digital map shape points, are not effective.

To be used for control-oriented ADAS, limitations of digital map databases have to be considered. For this purpose, several solutions are possible: to improve the digital map database accuracy, to use additional sensors, etc. Here, the digital map database is used as a virtual sensor for a trajectory generation which is devoted to the selection of the best suited trajectory in a set of possible solutions, i.e. a validity area. The latter is defined relative to the digital map database inaccuracies.

2.2 ADAS Global Structure

The trajectory generation process is part of an architecture dedicated to navigation-based control-oriented ADAS. In Fig.3, the different elements of a classic navigation system can be found (*GPS Receiver*, *Map-Matching Algorithm* and *Digital Map Database*). *Map-Matching* helps in the determination of the *Electronic Horizon* which constitutes the source of information providing the necessary data for:

- *Situation Classification*. This divides the current and upcoming road into several situations such as straight lines, bends, tunnels, etc. and so, improves the global view of the road context by providing context-aware information.
- *Road Model Estimation*. Using the information stored in the *Digital Map Database* and relative to the middle of the road, an estimation of the road profile (left and right lanes) is performed. This helps to define the different constraints required for the *Trajectory Generation* process.
- *Constraints Definition*. Considering the information provided by the *Road Model Estimation* and by vehicle sensors, this process defines the constraints which are related to the road, to the vehicle and to the driver. They constitute the information required by the *Trajectory Generation* process.
- *Trajectory Generation*. It generates constrained trajectories via a specific convex optimization approach. These trajectories have the advantage to be smooth and curvature continuous (with minimized curvature).

These four processes are the core of this study as they provide more complete data for control-oriented ADAS.

2.3 Trajectory Generation: overview

Trajectory generation originated in the robotic domain, and more precisely on studies focused on autonomous wheeled robots. The first studies defined trajectories as a succession of straight lines and arc-circles (Dubins (1957)). Even if this is a simple way to generate trajectories, it has been proved that it was not suitable for wheeled robots as it does not provide curvature continuity. An improvement to this solution was found through the introduction of trajectory portions which have a polynomial curvature expression between arc-circles and straight lines (Nagy and Kelly (2001)). For example, in Fraichard and Scheuer (2004), clothoids which provide polynomial variations of curvature along their arc-length, are a good solution for junctions. However, if this method solves the problem of curvature continuity for arc-line junctions, it uses three different mathematical models whose junctions have to be correctly defined. More generally, trajectories which are based on polynomial curvature expression have the advantage to directly act on the curvature representation and so, to provide at least its continuity. However, this method requires a numerical double integration of the curvature expression along the arc-length to get back to the trajectory coordinates. Consequently, errors are introduced into the trajectory generation process.

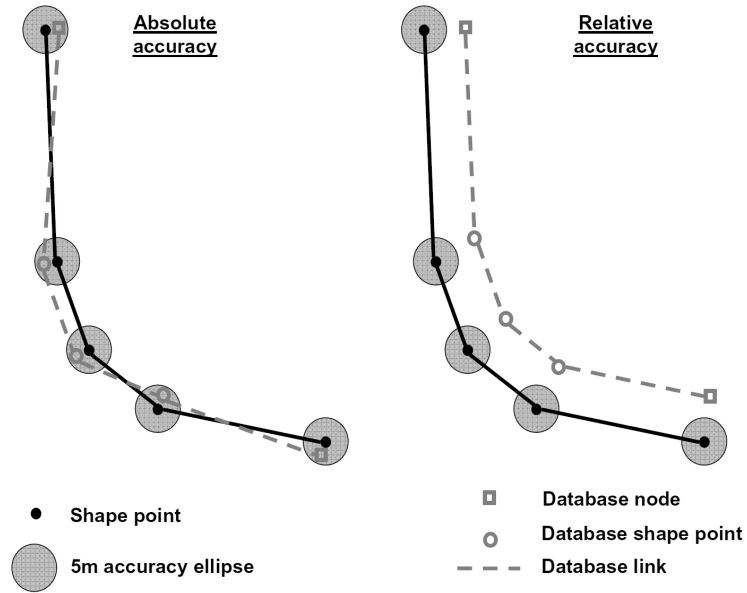


Figure 2 Digital Map Database Inaccuracy

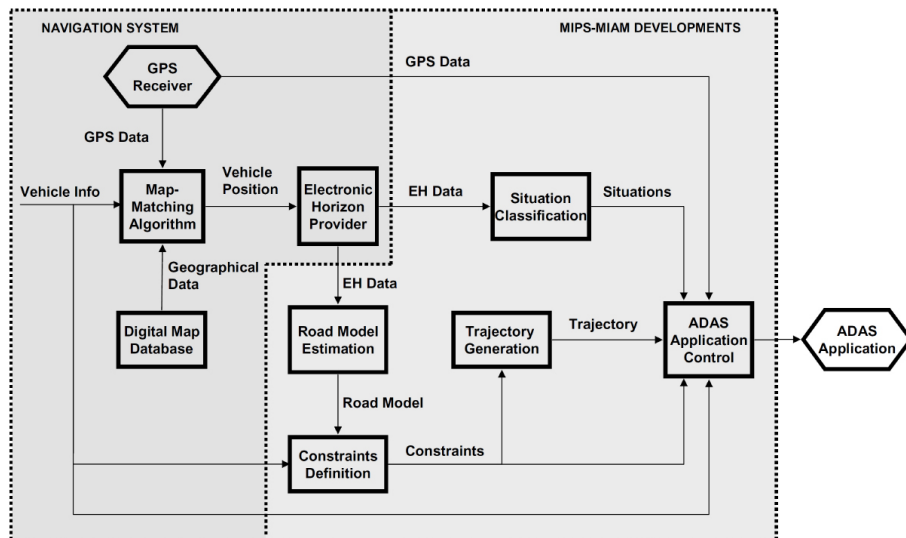


Figure 3 ADAS Control Application Architecture

In the last decade, increased computer power allowed more complex mathematical models to be used. One of these models is the Spline model (De Boor (1978)). A Spline is a piecewise polynomial interpolation whose junctions are constrained by first and second derivatives continuities. This allows the generation of smooth curvature continuous trajectories with low degree polynomials, and low calculation time for a large number of interpolated points. Well known Spline methods are Cubic Splines and Parametric Cubic Splines. Parametrization gives more freedom to the trajectory shape as it allows the definition of multi-dimensional Splines. Another Spline, often used in the literature, is the B-Spline model (Gomez et al. (2008)). The latter is based on the same principle as the Cubic Spline: a piecewise interpolation. The difference lies in the fact that B-Splines use Bezier curves in each interval. However, B-Splines require the definition of additional control points for all intervals. This implies an explosion of the point number which must be defined to get a trajectory.

Another mathematical model is the polar polynomial model (Nelson (1989)). This type of polynomial is defined in the polar referential and provides curvature continuous trajectories which give good results in bends. As for Cubic Splines, they use low order polynomials which are obtained according to continuity conditions on the first derivative, the slope and the curvature of the trajectory. This model has been successfully used for wheeled rolling systems trajectory generation (Pinchard et al. (1996)) or for trajectory modelling of different types of automotive vehicle drivers (Lauffenburger et al. (2003)). In addition, the use of such polynomials requires the definition of the polar referential and translation between polar and Cartesian coordinates. Furthermore, if polar polynomials are well suited for bends, they are not adapted to straight lines and are then often associated to quintic Cartesian polynomials (Nelson (1989)).

Among the presented models, the Parametric Cubic Spline has been chosen. Indeed, this mathematical model provides smooth curvature-continuous trajectories adapted to various road situations with no oscillations and no long calculation time.

3 Constrained Trajectory Generation

3.1 Problem Statement

The present objective is to provide a reference trajectory which satisfies several constraints linked to the Wheeled Rolling System (*WRS*), the road to be followed and finally the driver by considering control-based constraints during the generation. This allows the definition of a reference path which suits the limitation of the controlled system. Fig.4 describes the situation to be solved: giving a starting configuration $q_0 = (x_0, y_0, \theta_0, \kappa_0, \dot{\kappa}_0) \in \mathbb{R}^5$ defined by the *WRS*'s Centre of Gravity (*CoG*) position (x_0, y_0) , the *WRS*'s orientation (θ_0) , an initial curvature κ_0 of the path to be followed by the *CoG*, its respective derivative $\dot{\kappa}_0$ and a final configuration $q_n = (x_n, y_n, \theta_n, \kappa_n, \dot{\kappa}_n)$, an optimization technique defines the reachable path regarding the considered cost criterion. A trajectory is a continuous sequence of reachable configurations $(x, y, \theta, \kappa, \dot{\kappa})$ defined with respect to the constraints to be verified: normal driving conditions imply that the *WRS*

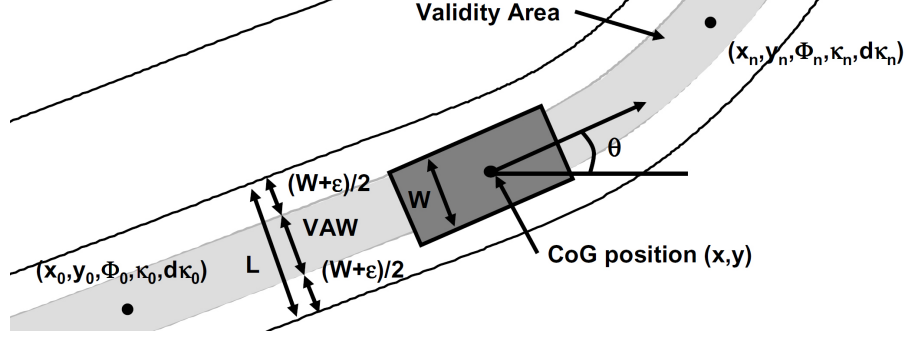


Figure 4 Vehicle in the Lane

stays in its current road lane. For car-like vehicles, this area is defined by the road on which the vehicle is located. Consider the situation described in Fig.4 which represents the vehicle in its driving lane with a total width of L . The reachable driving area, considering the absolute and relative inaccuracies of the digital map (cf. Fig.2) is then defined as $VAW = L - (W + \epsilon)$ with W the vehicle width and ϵ the digital map inaccuracies. Finally, in the proposed approach, the road is considered to be flat and only 2-dimensional parametric paths $(x(t), y(t)) \in \mathbb{R}^2$ are computed. The sequence of reachable configurations is defined based on constraints of different types (geometric, dynamic and kinematic).

Fig.5 is a schematic representation of the strategy used which is divided into two parts: the *Road Model Estimation* and the *Trajectory Generation*. The *Road Model Estimation* uses the shape points which are extracted from the EH. However, the latter only gives information about the road centreline. To generate the road boundaries, the first step is to use a geometrical translation which estimates the road boundary points w.r.t. the information of the digital map (lane width, number of lanes, etc.). The estimated points are then used by a Spline algorithm which computes continuous road boundaries and so, provide the required information for the *Trajectory Generation* process. This process first uses the road model and the vehicle width to define the trajectory validity area (cf. Section 3.2.1). This validity area is then used as a template for the optimization process which also includes the other aforementioned constraints (cf. Section.3.2) and which minimizes the trajectory strain energy.

3.2 Multiple Constrained Path Generation

WRS-like cars are known to be non-holonomic systems: not all the solutions of the configuration space are possible and limitations in the directions of motion have to be processed (Fraichard and Scheuer (2004)). An effective path generation algorithm should consider these limitations when computing the nominal trajectory the vehicle must follow.

This study considers three types of constraints. The first ones are geometric constraints linked, on the one hand, to the configuration space in which the *WRS* moves (to keep the vehicle in a prescribed driving area) and on the other hand, to the mechanical limits of the *WRS*. Due to the non-holonomy property of *WRS*, these vehicles are subject to kinematic constraints. This second type

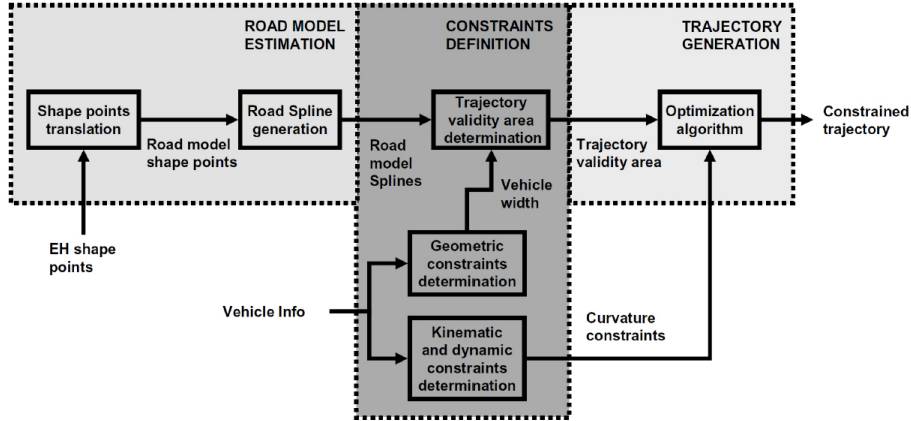


Figure 5 Trajectory Generation Strategy

of constraints involves (linear and rotational) velocities and acceleration limits of the kinematic variables describing the *WRS* (see Graettinger and Krogh (1989) for a description of these limits in the robotic domain). Finally, the third category of constraints are dynamic constraints which characterize the performance of the vehicle (braking/accelerating capabilities, maximum lateral acceleration, tyre/ground interaction etc.) and the actuators performance. They restrict the overall performance of the *WRS*.

3.2.1 Geometric Constraints

The geometric constraints are of two types: limitations related to the configuration space and limitations related to the mechanical conception of the *WRS*. The first constraints define the area prescribed in the configuration space, i.e. the set of possible configurations, in which the *WRS* is allowed to move. This is represented by the validity area in Fig.4.

Concerning the mechanical limitations of *WRS*, the limitations of the steering system imply a minimum break radius of the vehicle R_{min} . This turning radius is lower-bounded and consequently, the trajectory instantaneous curve radius must be constantly greater than the lower bound vehicle turning radius:

$$\frac{1}{R_{trajectory}} = \kappa_{trajectory} < \kappa_{max1} = \frac{1}{R_{min}} \quad (1)$$

Finally, in order to avoid any steering function discontinuities and since the steering angle is directly dependent on the instantaneous curvature of the trajectory to be followed, a continuous-curvature trajectory is necessary. So, the generated trajectories must be C^2 continuous.

3.2.2 Kinematic Constraints

The kinematic constraints depend on the speed profile along the trajectory. The main kinematic constraint to be considered for *WRS* is the limitation of the steering velocity. It is well known that the dynamic behaviour of the steering

system is upper bounded. Since the steering velocity is related to the derivative of the curvature, this implies that the trajectory instantaneous curvature derivative is upper-bounded:

$$\dot{\kappa}_{trajectory} < \dot{\kappa}_{max} \quad (2)$$

3.2.3 Dynamic Constraints

These constraints are due to the limited and often nonlinear dynamic behaviour of the *WRS* and its subsystems (bounded acceleration capabilities, variable ground/wheel interaction, etc.). They mainly influence the longitudinal and lateral accelerations and thus the velocities of the *WRS*. In order to provide safe trajectories, the maximum centrifugal acceleration allowed for curve negotiation is considered here. Different models which link the centrifugal acceleration to the trajectory curvature are available. This paper focuses on a simplified model given by:

$$\kappa_{max2} = \frac{\Gamma_{max}}{v^2} \quad (3)$$

with Γ_{max} the maximum allowed lateral acceleration and v the vehicle speed. Note that in the automotive domain, the driver is sensitive to the accelerations and mainly to the lateral acceleration. Considering a maximum value of acceleration Γ_{max} , driver-dependent factors are taken in account.

This relation implies that the path curvature must be upper bounded in order to ensure a limited centrifugal acceleration:

$$\kappa_{trajectory} < \kappa_{max2} \quad (4)$$

Considering the different elements presented in this section, the constraints upon the curvature can be summarized as follows:

$$\begin{aligned} \kappa_{trajectory} &< \min \{ \kappa_{max1}, \kappa_{max2} \} \\ \dot{\kappa}_{trajectory} &< \dot{\kappa}_{max} \end{aligned} \quad (5)$$

3.3 Parametric Cubic Spline Theory

As presented in Section 2.3, the Parametric Cubic Spline model has been chosen among all the different mathematical models for the generation of 2-dimension trajectories. A 2-dimension Parametric Cubic Spline is composed of two Cubic Splines:

$$f_i(t) = f(f_{x_i}(t), f_{y_i}(t), t) \begin{cases} f_{x_i}(t) = a_{f_{x_i}} t^3 + b_{f_{x_i}} t^2 + c_{f_{x_i}} t + d_{f_{x_i}} \\ f_{y_i}(t) = a_{f_{y_i}} t^3 + b_{f_{y_i}} t^2 + c_{f_{y_i}} t + d_{f_{y_i}} \end{cases} \quad (6)$$

with $t \in [t_i, t_{i+1}]$, $i = 1, 2, \dots, n-1$, n the number of interpolated points, $a_{f_{x_i}}$, $b_{f_{x_i}}$, $c_{f_{x_i}}$ and $d_{f_{x_i}}$, coefficients of x Cartesian coordinate Spline, $a_{f_{y_i}}$, $b_{f_{y_i}}$, $c_{f_{y_i}}$ and $d_{f_{y_i}}$, coefficients of y Cartesian coordinate Spline and t the parameter.

To ensure the smoothness of the trajectory and the continuity of the curvature, Cubic Splines are calculated on conditions: each point to be interpolated represents

a position condition. Moreover, continuity conditions on the first and the second derivatives of the trajectory at each point must be provided, by solving a linear equation system which computes the second derivative values (De Boor (1978)).

However, for one set of interpolated points, an infinite number of Splines is available. These Splines are defined depending on two major elements: *boundary conditions* and parameter values. As they can have strong effects on the Spline shape, they have to be correctly predefined.

The definition of the continuity constraints an interpolation Spline curve must satisfy at shape points is a crucial element. In the present solution, first and second derivative continuities have been chosen. However, for the first and the last points of the set of interpolated points, these constraints must be pre-defined. Numerous *boundary conditions* are possible: equal to zero (*natural conditions*), arbitrarily defined (*generalized conditions*), etc. (De Boor (1978)). A study on the different *boundary conditions* solutions and on their effect on Spline interpolation is available in Daniel et al. (2009). It shows that the Spline using the *boundary conditions* which have been developed in this study has the best efficiency compared to real road data.

Parametrization gives more freedom to the Spline, but it also adds another variable to the computation process - parameter t - whose values have to be chosen appropriately. Indeed, the parameter expression has strong effects on the trajectory shape as described in Floater (2008). Parameter values can be defined by:

- Linear relation:

$$t_i = i \text{ with } i \in [0, n-1] \in \mathbb{N} \quad (7)$$

- Chordal relation (distance dependent):

$$t_{i+1} = \sum_{i=0}^{n-2} \left((X_{i+1} - X_i)^2 + (Y_{i+1} - Y_i)^2 \right)^{\frac{1}{2}} \text{ with } t_0 = 0 \quad (8)$$

- Centripetal relation (square root of the distance dependent):

$$t_{i+1} = \sum_{i=0}^{n-2} \left((X_{i+1} - X_i)^2 + (Y_{i+1} - Y_i)^2 \right)^{\frac{1}{4}} \text{ with } t_0 = 0 \quad (9)$$

with X_i and Y_i the Cartesian coordinates of the i^{th} point to be interpolated.

To select the appropriate solution, these three different parameter definitions have been tested based on the race track used in Section 5 (cf. Fig.6). The real road data are based on centimetric accuracy measurements (provided by the race track builder). The different Splines generated, based on the centreline shape points of this race track, are compared to this accurate representation of the real road, and the results are presented in Table.1.

This table confirms the performance of the Spline which uses centripetal values. Indeed, the difference with the real road data is greatly reduced compared with the Spline defined with chordal values. However, the improvement compared with the Spline defined with linear parameter values is low: the Spline defined with linear values is almost as efficient as the Spline defined with centripetal parameter values.

Table 1 Impact of parameter definition on Spline efficiency

<i>Parameter values</i>	<i>Linear</i>	<i>Chordal</i>	<i>Centripetal</i>
Maximum distance (m)	0.63	1.15	0.56
Mean distance (m)	0.26	0.34	0.24

4 Trajectory Generation Formulated as an Optimization Problem

4.1 Formalization

This Section describes the integration of the different constraints (presented in Section.3.2) into the selected mathematical model. Considering that Splines have polynomial expressions, the goal of the present trajectory generation process is to find the optimal coefficients $a_{f_{S_i}}$, $b_{f_{S_i}}$, $c_{f_{S_i}}$ and $d_{f_{S_i}}$ with: $S_i = [x_i, y_i]^T$, $i = 0, 1, \dots, n-1$ for the constrained trajectory $f_i(t)$ (cf. 6) with t the parameter defined in $[t_i, t_{i+1}]$ and n the number of considered points.

Based on the results presented in Section 3.3, linear parameter values are used. This avoids the non-linear expression related to the centripetal expression and generated good results. The different intervals $[t_i, t_{i+1}]$ consequently corresponds to: $[0, 1], [1, 2], \dots, [t_{n-1}, t_n]$ so that $t_{i+1} - t_i = 1, \forall i \leq n-1$. This problem is here considered as referring $n-1$ times to the problem of defining $f_i(t)$ on the $[0, 1]$ interval.

4.1.1 Inequality Constraints

Let $g_i(t)$ and $e_i(t)$ be the two boundary curves which describe the validity area. The optimization consists in finding a Spline $f_i(t)$ such that:

$$e_i(t) \leq f_i(t) \leq g_i(t) \quad (10)$$

with:

$$\begin{aligned} g_i(t) & \{ g_{S_i}(t) = a_{g_{S_i}} t^3 + b_{g_{S_i}} t^2 + c_{g_{S_i}} t + d_{g_{S_i}} \\ e_i(t) & \{ e_{S_i}(t) = a_{e_{S_i}} t^3 + b_{e_{S_i}} t^2 + c_{e_{S_i}} t + d_{e_{S_i}} \end{aligned} \quad (11)$$

To generate a Spline which is constricted by the validity area boundaries, the difference between the desired Spline and the upper and lower bounds must be negative or positive respectively:

$$\begin{aligned} f_i(t) - g_i(t) & \leq 0 \\ f_i(t) - e_i(t) & \geq 0 \end{aligned} \quad (12)$$

The question raised by (12) concerns more generally Spline positivity conditions regarding its coefficients. The simplest way to ensure the positivity of a cubic polynomial of the form $f(t) = at^3 + bt^2 + ct + d$ with $t \in [0, 1]$ is when all the coefficients a , b , c and d are positive. However, this solution is highly conservative and strongly restricts the set of possible polynomials. In the literature, several studies have been carried out on the positivity of cubic polynomials. Particularly,

Schmidt showed in Schmidt and Heß (1988) that a cubic polynomial $f(t)$ defined for $t \in [0, 1]$ can be equivalently transformed into:

$$\hat{f}(s) = \alpha s^3 + \beta s^2 + \gamma s + \delta \quad (13)$$

with $s \in [0; +\infty[$, $t = \frac{s}{(1+s)}$ and with:

$$\begin{cases} \alpha = a + b + c + d \\ \beta = b + 2c + 3d \\ \gamma = c + 3d \\ \delta = d \end{cases} \quad (14)$$

The positivity of (13) is then defined regarding the α , β , γ and δ coefficients such as:

$$\begin{cases} \hat{f}(s) \geq 0 \Rightarrow (\alpha, \beta, \gamma, \delta) \in A \cup B, \text{ with} \\ A = \{(\alpha, \beta, \gamma, \delta) : \alpha \geq 0, \beta \geq 0, \gamma \geq 0, \delta \geq 0\} \\ B = \{(\alpha, \beta, \gamma, \delta) : \alpha \geq 0, \delta \geq 0, 4\alpha\gamma^3 + 4\delta\beta^3 + 27\alpha^2\delta^2 - 18\alpha\beta\gamma\delta - \beta^2\gamma^2 \geq 0\} \end{cases} \quad (15)$$

A and B are the two solution sets for the different coefficients. The solution subset A presents linear constraints and so, can easily be adapted to the present context. Subset B is defined by a non linear expression involving that the set $A \cup B$ is rather complicated to be used in practice. In the present formulation of the optimization process, the considered subset of possible solutions is limited to A .

Based on (13), the inequalities of the parameters a , b , c and d for $f(t)$ are then defined for $t \in [0, 1]$ such that:

$$\begin{aligned} f(t) \geq 0 &\Rightarrow (a, b, c, d) \in A, \text{ with} \\ A &= \{(a, b, c, d) : a + b + c + d \geq 0, b + 2c + 3d \geq 0, c + 3d \geq 0, d \geq 0\} \end{aligned} \quad (16)$$

The combination of (12) and (16) leads to the set of linear inequalities presented in (17). It can be noticed that the inequalities on $d_{f_{S_i}}$ define a square of possible solutions. These inequalities are directly related to the position of the Spline points. In particular cases, this may lead to bound values which are very close, so that $d_{f_{S_i}}$ can only vary by a few centimetres. The considered inequalities can thus be very conservative.

$$\begin{cases} a_{e_{S_i}} + b_{e_{S_i}} + c_{e_{S_i}} + d_{e_{S_i}} \leq a_{f_{S_i}} + b_{f_{S_i}} + c_{f_{S_i}} + d_{f_{S_i}} \leq a_{g_{S_i}} + b_{g_{S_i}} + c_{g_{S_i}} + d_{g_{S_i}} \\ b_{e_{S_i}} + 2c_{e_{S_i}} + 3d_{e_{S_i}} \leq b_{f_{S_i}} + 2c_{f_{S_i}} + 3d_{f_{S_i}} \leq b_{g_{S_i}} + 2c_{g_{S_i}} + 3d_{g_{S_i}} \\ c_{e_{S_i}} + 3d_{e_{S_i}} \leq c_{f_{S_i}} + 3d_{f_{S_i}} \leq c_{g_{S_i}} + 3d_{g_{S_i}} \\ d_{e_{S_i}} \leq d_{f_{S_i}} \leq d_{g_{S_i}} \end{cases} \quad (17)$$

4.1.2 Equality Constraints

In addition to the inequalities, the optimization process must fulfil the continuity requirements. Consequently, it must provide C^0 , C^1 and C^2 continuities which are obtained, for Parametric Cubic Splines, via the following relation:

$$\begin{cases} f_{S_i}(t_{i+1}) = f_{S_{i+1}}(t_{i+1}) \\ \dot{f}_{S_i}(t_{i+1}) = \dot{f}_{S_{i+1}}(t_{i+1}) \\ \ddot{f}_{S_i}(t_{i+1}) = \ddot{f}_{S_{i+1}}(t_{i+1}) \end{cases} \Rightarrow \begin{cases} a_{f_{S_i}} + b_{f_{S_i}} + c_{f_{S_i}} + d_{f_{S_i}} = d_{f_{S_{i+1}}} \\ 3a_{f_{S_i}} + 2b_{f_{S_i}} + c_{f_{S_i}} = c_{f_{S_{i+1}}} \\ 6a_{f_{S_i}} + 2b_{f_{S_i}} = 2b_{f_{S_{i+1}}} \end{cases} \quad (18)$$

4.2 Cost Criterion

The cost criterion can be defined regarding various elements or parameters. Common criteria are:

- *No criteria.* The solution of the optimization problem only considers the constraints. In the current context, this kind of optimization leads to the generation of a trajectory which only fulfils the geometric constraints given by the validity area boundaries. Results based on this strategy will be presented in Section 5.
- *Trajectory length.* This criterion is used in numerous robotic applications as it allows the definition of the shortest trajectory which links the current robot configuration to the desired robot configuration. For example, it can be described according to the trajectory point coordinates x and y in the following way:

$$D = \int f(x(t), y(t), t) ds \quad (19)$$

- *Trajectory energy.* In the proposed constrained trajectory generation, the strain energy criterion is expressed according to the trajectory curvature κ as follows (Delingette et al. (1991)):

$$E = \int \kappa^2 ds \quad (20)$$

In this particular case, the curvature κ corresponds to:

$$\kappa(x(t), y(t), t) = \frac{\ddot{y}(t)\dot{x}(t) - \ddot{x}(t)\dot{y}(t)}{(\dot{x}^2(t) + \dot{y}^2(t))^{\frac{3}{2}}} \quad (21)$$

This criterion is in accordance with the aforementioned constraints. In addition, its minimization implies smoothing the trajectory curvature which is directly linked to the energy consumption of the vehicle. For example, a trajectory tracking system, with a smoother curvature, will require less steering energy. Finally, this criterion is well adapted to the current global context of power saving and power consumption reduction.

By replacing $x(t)$ and $y(t)$ functions into their formal expressions, (21) becomes non-linear (and so becomes (20)). The optimal solution (20) is also hard to formulate and to compute (Ye and Qu (1999)). To overcome this problem, a suboptimal solution is determined. Considering the expression of $x(t) = f_{x_i}(t)$ and $y(t) = f_{y_i}(t)$, the objective is to simultaneously minimize the curvature of each parametric curve κ_x and κ_y given by:

$$\kappa_x(x(t), t) = \frac{\ddot{x}(t)}{(1 + \dot{x}^2(t))^{\frac{3}{2}}} \text{ and } \kappa_y(y(t), t) = \frac{\ddot{y}(t)}{(1 + \dot{y}^2(t))^{\frac{3}{2}}} \quad (22)$$

Considering that $\dot{x}(t)^2$ and $\dot{y}(t)^2$ are small compared with 1, the energy of the suboptimal solution which corresponds to the cubic interpolation Spline with C^2 continuity, can be expressed by:

$$\tilde{E} = \int_{t_0}^{t_n} (\dot{x}^2 + \dot{y}^2) dt \quad (23)$$

Regarding the Spline expression (cf. (6)) and its continuity properties, the continuous expression of the energy \tilde{E} can be discretized using the classic Euler method. Finally, according to Pollock (1999), the total is assumed to be of the form:

$$\tilde{E} = \sum_{i=1}^{n-2} \frac{4h_i}{3} \left(b_{f_{x_i}}^2 + b_{f_{x_i}} b_{f_{x_{i+1}}} + b_{f_{x_{i+1}}}^2 \right) + \sum_{i=1}^{n-2} \frac{4h_i}{3} \left(b_{f_{y_i}}^2 + b_{f_{y_i}} b_{f_{y_{i+1}}} + b_{f_{y_{i+1}}}^2 \right) \quad (24)$$

4.3 Optimization

The trajectory generation presented in this section and formulated as an optimization problem must fulfill the following conditions:

- Linear inequality constraints (cf. (17)).
- Linear equality constraints (cf. (18)).
- A discretized quadratic cost criterion (cf. (24)).

There are several optimization techniques which help to solve such systems (Boyd and Vandenberghe (2004)). However, considering the different types of constraints and the fact that they all have a linear or quadratic expression, the convex quadratic programming optimization approach has been chosen; it is of the following form:

$$\begin{aligned} \Theta^* &= \min_{\Theta} \frac{1}{2} \Theta^T H \Theta + f^T \Theta \\ \text{such that: } &\begin{cases} A\Theta = B \\ C\Theta \leq D \end{cases} \end{aligned} \quad (25)$$

with $\Theta \in \mathbb{R}^{[2 \cdot 4 \cdot (n-1)] \times [1]}$ the optimal coefficients of $f_i(t)$ interpolating n points such that:

$$\Theta = [a_{f_{S_1}}, b_{f_{S_1}}, c_{f_{S_1}}, d_{f_{S_1}}, \dots, a_{f_{S_{n-1}}}, b_{f_{S_{n-1}}}, c_{f_{S_{n-1}}}, d_{f_{S_{n-1}}}]^T \quad (26)$$

and $A \in \mathbb{R}^{[2 \cdot 3 \cdot (n-2)] \times [2 \cdot 4 \cdot (n-1)]}$, $B \in \mathbb{R}^{[2 \cdot 4 \cdot (n-2)] \times [1]}$, $C \in \mathbb{R}^{[2 \cdot 4 \cdot 2 \cdot (n-1)] \times [2 \cdot 4 \cdot (n-1)]}$, $D \in \mathbb{R}^{[2 \cdot 4 \cdot 2 \cdot (n-1)] \times [1]}$.

To solve (25), the well known *Dantzig-Wolfe* algorithm has been used here. It is clear that this approach is well suited for the present problem since:

- The equality and inequality constraints can respectively be written in the $A\Theta = B$ and $C\Theta \leq D$ matrix form,

- The energy criterion, due to its discrete formulation, is only expressed using quadratic terms of the different $b_{f_{x_i}}$ and $b_{f_{y_i}}$ coefficients. It can also be easily expressed in the matricial expression $\Theta^T H \Theta$ but, as there are only quadratic elements, $f^T \Theta = 0$,
- This optimization is convex quadratic, so there is only one global minimum,
- It does not need long calculation time which may coincide with real-time constraints. Indeed, this constrained trajectory generation has been tested in real-time conditions with a standard navigation system using a $1Hz$ sampling frequency. In these conditions, the system is able to compute a new constrained trajectory over a sliding horizon of a least $100m$ in front of the vehicle. These results have been obtained using a Dual Core CPU computer with $4Gb$ RAM.

4.4 Remark

It can be noted that geometric constraints (limitations of the configuration space and continuity of the trajectory) are explicitly formulated in the optimization through (17) and (18) and that the kinematic and dynamic constraints are implicitly described by the criterion to be minimized (the minimization of the strain energy should provide low curvature variations and values). These hypotheses are post-checked after optimization. Note that the checking of the curvature derivative is based on Fraichard and Scheuer (2004) which, considering that the *WRS* moves on a plane surface with non-sliding wheels, defines $\dot{\kappa}$ for a constant velocity v such that:

$$\dot{\kappa} = \frac{\dot{\phi}}{vb \cos^2(\phi)} \quad (27)$$

with b the wheelbase of the vehicle and ϕ the wheel angle.

As $\dot{\phi} \leq \frac{\dot{\phi}}{\cos^2(\phi)} \forall \phi$, expression (27) can be simplified in a more conservative way:

$$\dot{\kappa} = \frac{\dot{\phi}}{vb} \quad (28)$$

5 Test Results

This section presents the results of the trajectory generation presented in Section 4. Further to the study of the road model estimation results, the comparison of three trajectory generation techniques on different roads is described.

5.1 Road Model Estimation Results

Since the road model is used to define the geometrical constraints of the optimization, it is important to validate the results obtained. The road model is composed of three continuous curves: the road middle curve, the left road boundary curve and the right road boundary curve. The two boundaries are

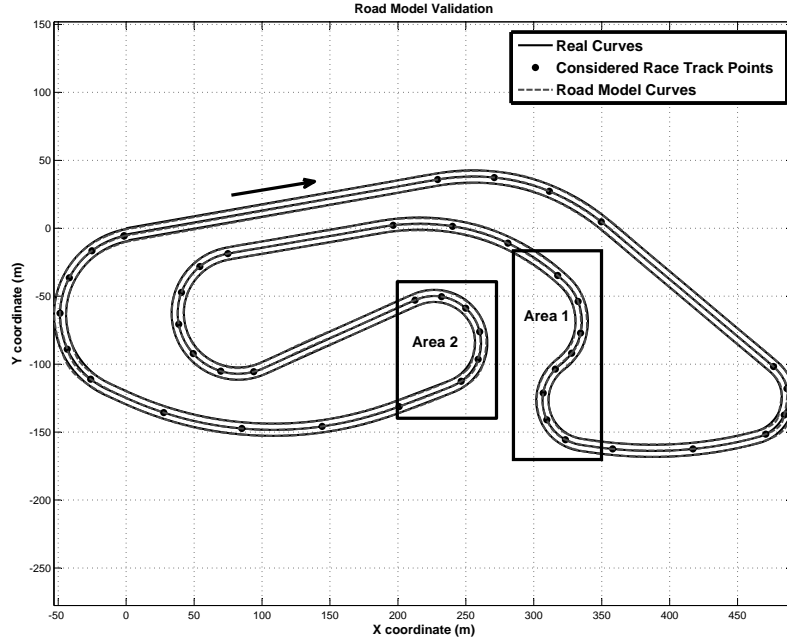


Figure 6 Complete Road Model Estimation

determined using the geometric translation of the middle road curve by several metres to the left and to the right of the road centreline.

To validate the road model estimation, comparison tests were carried out with real road data which correspond to a race track. These data were obtained via topological measurements with a centimetre accuracy. The goal is now to compare the real road data with the road model defined by Splines. Note that all the generated trajectories presented in this section have been computed using linear parameter values based on the shape points of a digital map database (black points in Fig.6). Fig.6 presents the results of the road model estimation along the entire race track. It is difficult to distinguish any difference between the real data and the Spline. This is confirmed by Fig.7 top plot which selects a portion of the race track. The squares *Area 1* and *Area 2* are specific road portions used to compare the different trajectories in the following section.

In Fig.7 bottom plot, the results of the error study between corresponding curves (real left curve against estimated left curve and real right curve against estimated right curve) show that the error is close to or lower than 50cm with a mean value of 29cm in both cases. This confirms that, with only the information of the digital map database, accurate results in the road profile reconstruction can be obtained considering precise shape point locations.

5.2 Constrained Trajectory Generation Results

For clarity reasons, the figures presented in the next Sections correspond to a focus on relevant parts of the race track. Only the validity area boundaries and the trajectories to be compared are represented (the road boundaries have

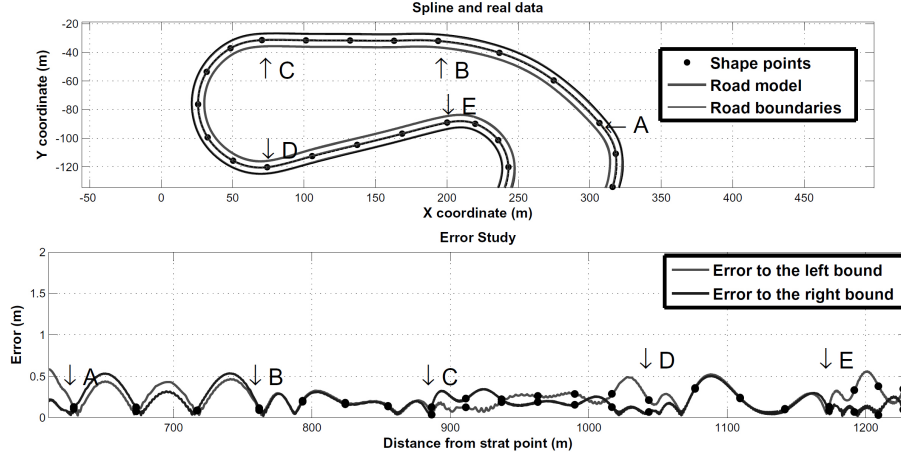


Figure 7 Road Model Estimation Focus

been removed). The considered driving lane is the right lane: the validity area is computed in this part of the road.

Fig.8 corresponds to results obtained in *Area 1*. This figure presents - on the one hand - the results of the constrained trajectory generated without energy minimization (*WOEM*) and - on the other hand - a Spline which interpolates shape points located in the centre of the lane (*CL*). Note that for clarity reasons, the figure has been rotated by 90° . First, it is important to notice that both trajectories are kept in the validity area and so, provide the geometric constraints linked to the road. Secondly, it is clear that there is hardly any difference between them: they are both located near to the middle of the lane. This is confirmed by the curvature comparison depicted in Fig.9. These figures tend to show that the *WOEM* trajectory (which only provides the constraints) gives results similar to those of the *CL* trajectory (normal Spline interpolation). However, the major difference lies in the approach: if the normal Spline interpolation requires the accurate location of the different shape points and so, defines a unique solution based on *a priori* knowledge, the constrained trajectory generation looks into all the possible trajectories allowed by the validity area and selects a trajectory. Consequently, the constrained trajectory is less sensitive to positioning errors and to digital map database errors. There is also less *a priori* information required.

5.3 Energy Minimization Results

As mentioned in Section 4.2, the trajectory energy is the cost criterion of the present constrained trajectory generation. It should help in the definition of trajectories which have better properties in terms of smoothness compared with those presented in the previous section. To validate this aspect, the present section is subdivided into two parts corresponding to the areas marked *Area 1* and *Area 2* in Fig.6.

Fig.10 presents the results of the *WOEM* trajectory and the trajectory which minimizes the energy (*WEM*) in *Area 1*. Contrary to the previous figure (Fig.8), the differences are here more visible as the *WEM* trajectory uses the available

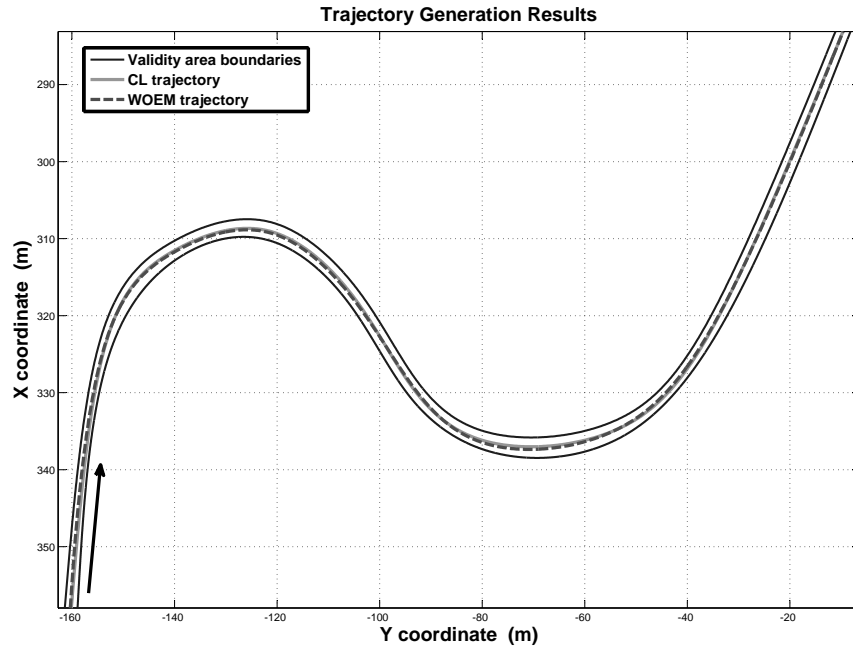


Figure 8 CL vs WOEM Trajectories in Area 1

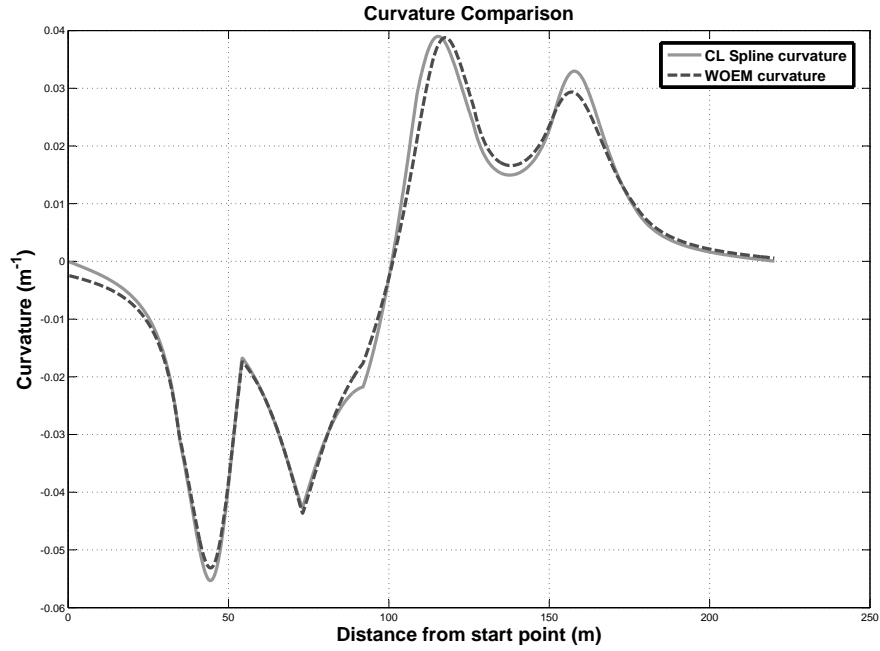


Figure 9 Curvature Comparison in Area 1

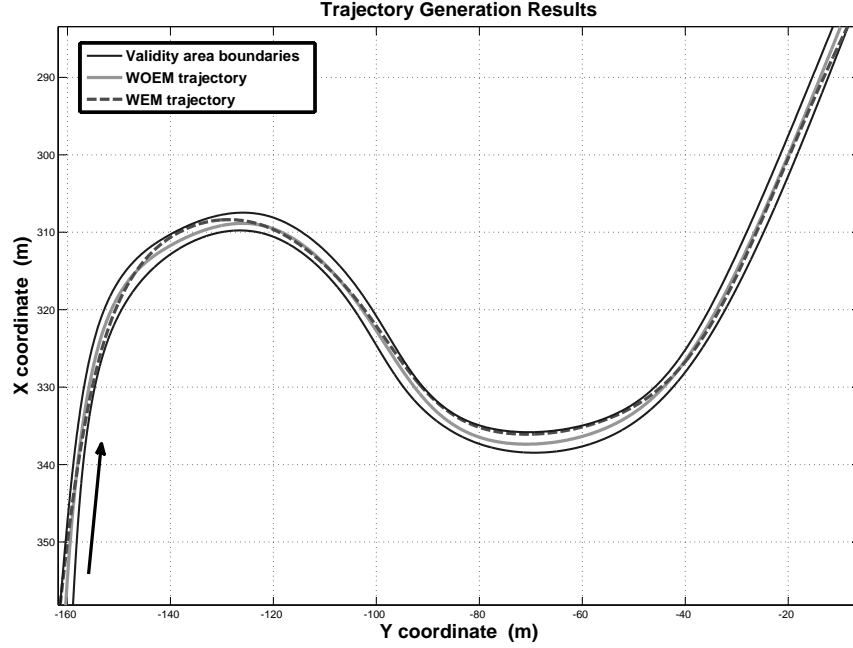
Figure 10 *WOEM vs WEM Trajectories in Area 1*

Table 2 Energy Comparison

	E_{CL}	E_{WOEM}	E_{WEM}	Eff_{CL}	Eff_{WOEM}
Area 1	0.0391	0.0386	0.0364	6.8%	5.6%
Area 2	0.0248	0.0241	0.0222	10.6%	7.9%

lateral area with greater efficiency than the *WOEM* trajectory: it is located alternately in the interior and in the exterior of the validity area. Note that both trajectories are still kept inside the validity area. The curvatures, depicted in Fig.11, also present more visible differences. Indeed, the curvature of the *WEM* trajectory has lower amplitudes and is smoother than the *WOEM* trajectory curvature, especially in the first bend (between 25m and 100m).

Fig.12 presents the *WOEM* and *WEM* trajectories in *Area 2*. Here again, both trajectories are within the validity area. Moreover, as for the previous test, the *WEM* trajectory uses the available lateral area more efficiently while the *WOEM* trajectory stays close to the centre of the lane. This is confirmed by the curvature plot (Fig.13) which shows an improvement of the curvature shape as curvature fluctuations are smoothed. Remember that the objective is not to estimate the real road curvature represented by a dashed/dotted line in Fig.9, Fig.11 and Fig.13 but to generate a bounded trajectory based on digital map data. However, even with the minimization of the energy, the *WEM* trajectory presents a peak at the end. This is mainly due to the shape of the validity area which does not correspond to reality on this particular section.

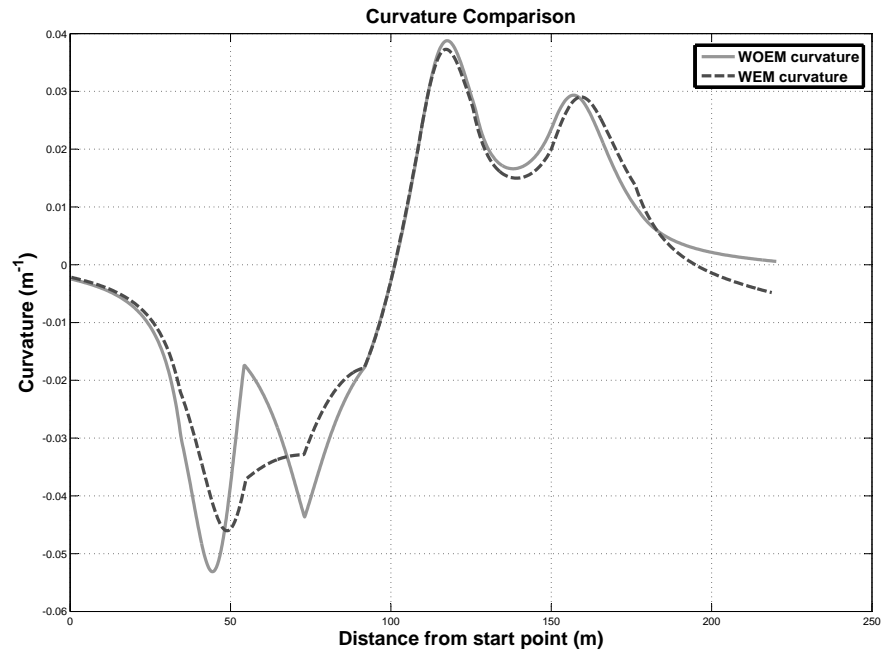


Figure 11 Curvature Comparison in Area 1

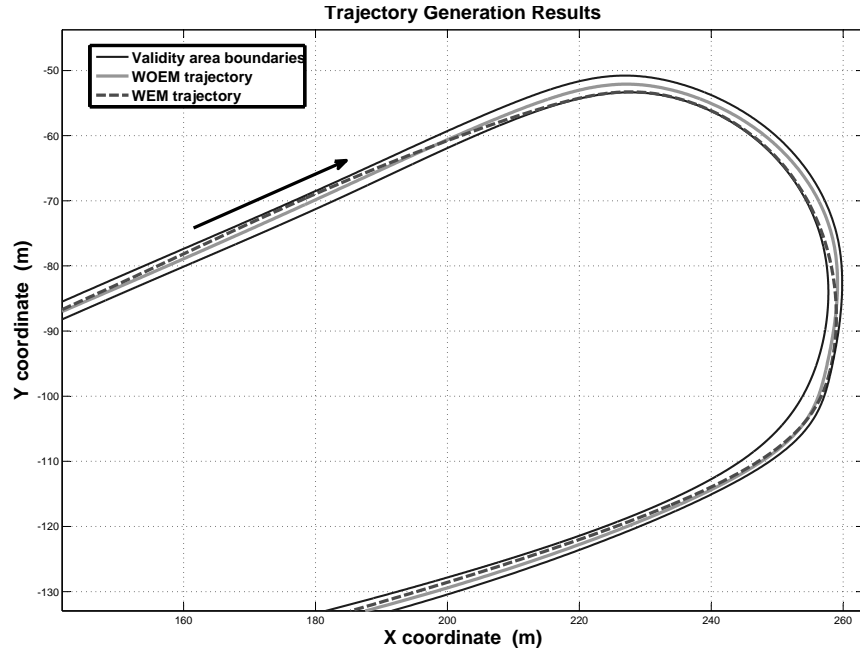


Figure 12 WOEM vs WEM Trajectories in Area 2

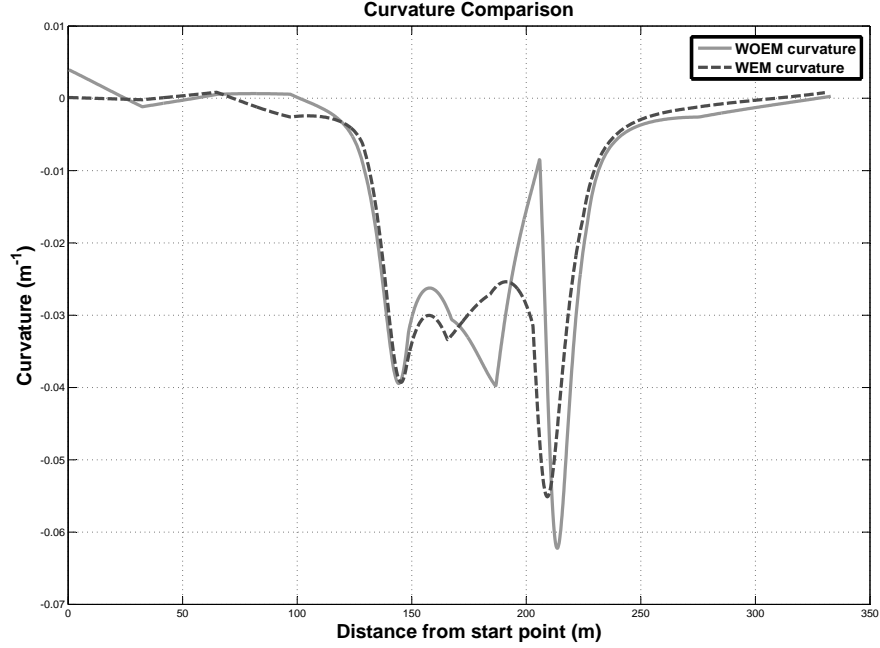


Figure 13 Curvature Comparison in *Area 2*

Table.2 presents the different energies computed respectively for the centreline trajectory (E_{CL}) and for the constrained trajectories generated without (E_{WOEM}) and with (E_{WEM}) the energy minimization in *Area 1* and *Area 2*. In addition, the reduction efficiency is described ($Eff_{CL} = \frac{E_{CL} - E_{WEM}}{E_{CL}}$ and $Eff_{WOEM} = \frac{E_{WOEM} - E_{WEM}}{E_{WOEM}}$). It must first be noted that the integration of the energy criterion has a positive impact on the energy value as it is reduced in each test. Then, the energy reduction is significant as it goes up to 10.0% in these cases. The effectiveness of the solution depends on the initial smoothness of the road geometry. In fact, the test results described in this section were obtained in circular bends. The *CL* and *WOEM* trajectories which are located near the centre of the validity area (so near the centre of the lane) are also already close to the minimum energy curve. This may explain the reduction values of 5.6% and 6.8% for the considered race track sections. Finally remember that the search area is limited to subset *A*. Better results could be expected, considering $A \cup B$.

The results presented in this section show the good properties of the trajectory generated under constraints with the minimization of the energy compared with real data. Indeed, the minimization of the energy provides trajectories which is always located in the validity area and so, in the current driving lane. Moreover, these trajectories make a more use more efficient use of the lateral area available in the validity area for energy reduction purposes. The advantage of this minimization lies in a smoothed curvature shape.

Table 3 Post-checking of the constraints for Area 1

	$k_{max}(m^{-1})$	$\dot{\kappa}_{max}(m^{-1}.s^{-1})$	$\Gamma_{max}(m.s^{-2})$
<i>CL</i>	0.055	0.005	3.8
<i>WOEM</i>	0.053	0.005	3.7
<i>WEM</i>	0.046	0.003	3.2
<i>Threshold</i>	0.090	0.038	3.0

Table 4 Post-checking of the constraints for Area 2

	$k_{max}(m^{-1})$	$\dot{\kappa}_{max}(m^{-1}.s^{-1})$	$\Gamma_{max}(m.s^{-2})$
<i>CL</i>	0.068	0.011	4.7
<i>WOEM</i>	0.062	0.010	4.3
<i>WEM</i>	0.055	0.006	3.8
<i>Threshold</i>	0.090	0.038	3.0

5.4 Post-checking of the Curvature and Curvature Derivative Constraints

As mentioned in Section 4.1, post-checking of the constraints on the curvature (cf. (1) and (3) with (4)) and on its derivative (cf. (2)) was carried out for each test. The results of this checking are available for *Area 1* and *Area 2* respectively in Table.3 and Table.4. A threshold of $0.09m^{-1}$ was taken for the constraint linked to the minimum curve radius of the car (1). It corresponds to an average turn radius of $11m$. The threshold of the constraint linked to the curvature derivative is of $0.038m^{-1}.s^{-1}$. It was obtained using (28) with a wheelbase b of $2.5m$, a speed v of $30km.h^{-1}$ (speed based on the configuration of the bends), and a maximum steering angle speed of $15.7rad.s^{-1}$ obtainable with an electrically driven steering wheel for lateral control (Pouly (2009)). Finally, the threshold of the constraint linked to the car acceleration (4) was fixed to the common lateral acceleration value used for comfortable driving: $3m.s^{-2}$. These three constraints are respectively marked k_{max} , $\dot{\kappa}_{max}$ and Γ_{max} in the tables. On the one hand, these tables clearly show that the first constraint which corresponds to the maximal curvature allowed by the car geometry, is respected by all trajectories. This is also the case for the second constraint which is linked to the maximal steering speed. On the other hand, the effect of the energy minimization can also be noticed in these tables. Indeed, the maximum curvature and the maximum curvature derivative of the *WME* trajectories are always reduced compared with the others.

However, the third constraint which is linked to the maximal allowed acceleration, is not provided by any trajectory for a fixed constant speed of $30km.h^{-1}$. Nevertheless, Fig.14 shows that this constraint is globally satisfied and that there is only a transient overshoot at the entrance of the first bend (around $50m$). Note that a small reduction of the speed, respectively $1km.h^{-1}$ and $4km.h^{-1}$ for *Area 1* and *Area 2*, would help to satisfy the constraint on the whole trajectory. Also note that if the third constraint is not provided, it can be due

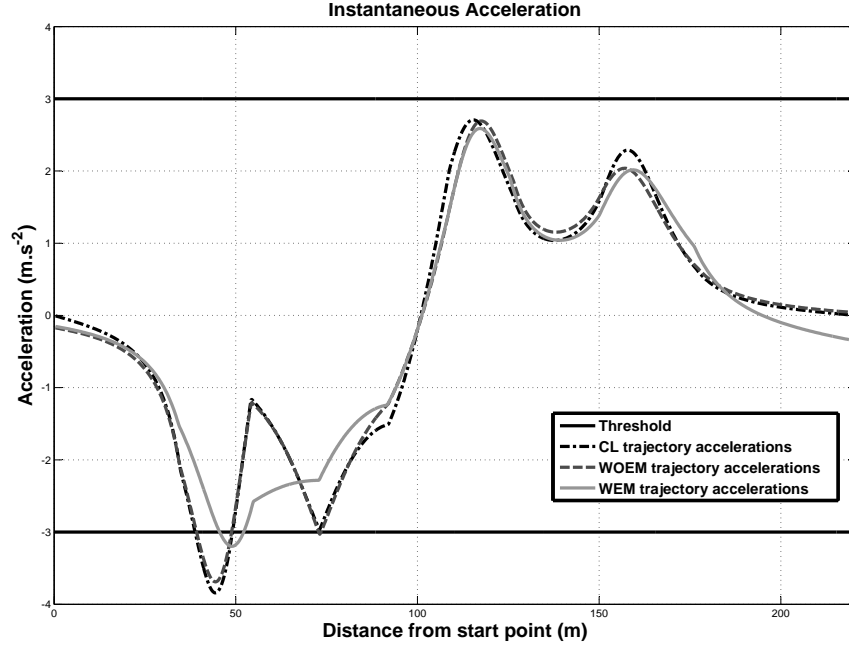


Figure 14 Trajectory Instantaneous Acceleration in *Area 1*

to the average definition of the validity area. In fact, for *Area 2*, the end of the second bend which presents a curvature peak, also presents a validity area of a shape which does not correspond to reality: the validity area width is not constant. The trajectory generated under constraint is also strongly dependent on the road geometry and on the quality of the validity area. Nevertheless, the benefits of the present trajectory generation are shown through the reduction of the curvature maximal value coupled to the reduction of its derivative maximum.

5.5 Using Classical Digital Map Data

In the previous sections, the comparison of three trajectory generation techniques was studied on a test track composed of regular bends separated by straight lines. This section is devoted to the comparison of trajectories on open roads digitalized with the inaccuracies described in Section 2.1. As for previous figures and for clarity reasons, only the boundary curves, the *WOEM* and the *WEM* trajectories are presented in Fig.15.

Similarly to the results presented in the previous sections, both trajectories are kept within the validity area and the *WEM* uses the area available more efficiently. The corresponding curvatures presented in Fig.16 show an important smoothing of the curvature. These figures show similar results and similar benefits of the *WEM* trajectory compared with conventional approaches (*CL* and *WOEM*). This is confirmed by Table.5 which, in addition to the energy values of the trajectories presented in Fig.15 (*Test 1*), shows the energy values for additional roads of different compositions. These roads correspond to a right bend (*Test 2*), a country

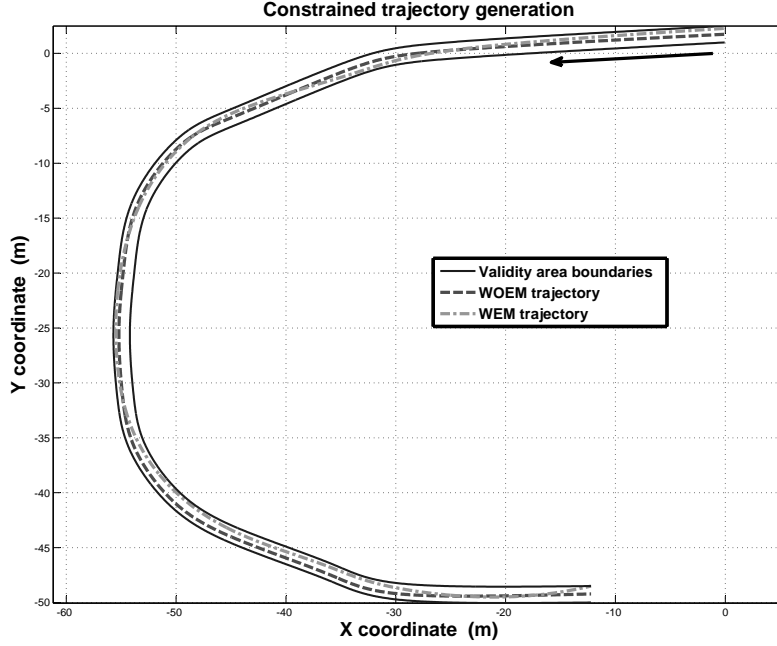


Figure 15 Trajectory Generation in a Non-regular Bend

Table 5 Energy Comparison

	<i>Test 1</i>	<i>Test 2</i>	<i>Test 3</i>	<i>Test 4</i>	<i>Test 5</i>
E_{CL}	0.0117	0.0223	0.0506	0.141	0.0354
E_{WOEM}	0.0119	0.0228	0.0507	0.145	0.0356
E_{WEM}	0.0106	0.0165	0.0452	0.113	0.0270
Eff_{CL}	9.1%	26.3%	10.5%	20.1%	23.9%
Eff_{WOEM}	10.5%	27.8%	10.8%	22.0%	24.4%

road (*Test 3*) and an in-city road composed of multiple bends and straight lines (*Test 4*), and finally a roundabout followed by a straight line (*Test 5*).

The integration of the energy criterion has a positive impact on the energy value as the energy is reduced in each test. Then, the energy reduction is variable as it passes from 9.1% to 27.8%, but in nearly each case, it is higher than the reduction values presented in the previous sections. This is due to the non-regular shape of the test considered here and to their diverse composition (succession of opposite direction bends, roundabouts and straight lines, etc.).

6 Conclusion

This document has presented a new approach to trajectory generation for control-oriented Advanced Driver Assistance Systems. Based on the inaccuracies of the digital map database, a template which defines the set of possible trajectories is

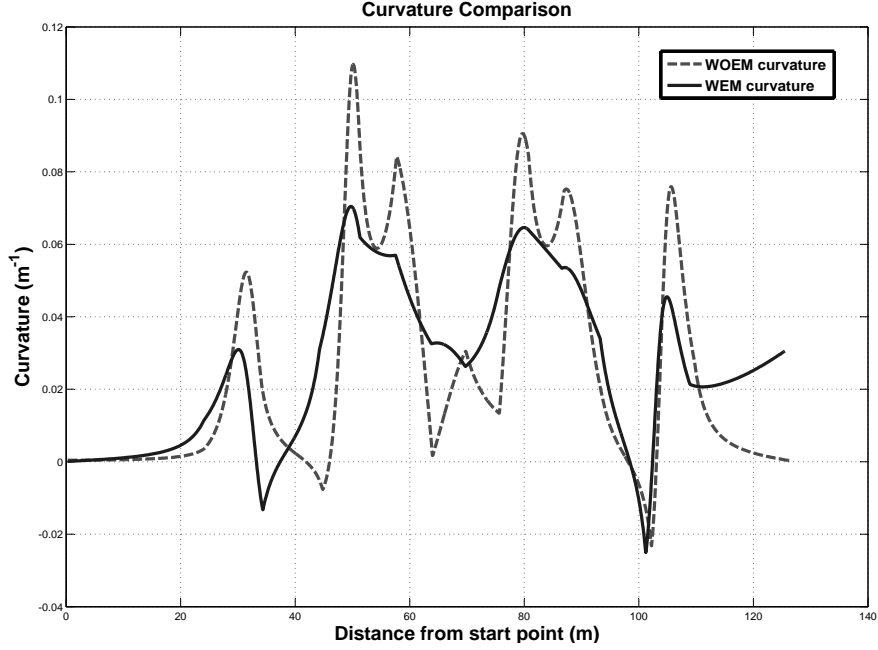


Figure 16 Curvature Comparison in a Non-regular Bend

generated. The trajectory generation, defined as an optimization problem, searches for a suboptimal solution, included in the template, which satisfies geometric, kinematic and dynamic constraints. The convex optimization is performed through the minimization of the trajectory strain energy.

Tests were carried out to validate the improvements of the present approach. Real data of a race track were used to compare three trajectory generation methods: basic Spline interpolation, constrained trajectory generation without energy minimization and finally constrained trajectory generation with energy minimization. The study of the different trajectory shapes, curvature and energy has shown that the present approach gives better results than the simple interpolation approach. In addition, the present constrained trajectory generation is a good solution for control-oriented ADAS as it provides a set of continuous information about the current and the upcoming road context, thus overcoming the described limitations of the digital map database. At the same time it directly integrates - into the trajectory generation process - control-based constraints related to the road geometry but also constraints related to the kinematic and dynamic properties of the car, which are commonly included in control systems.

Future research work will concern the integration of the dynamic and kinematic constraints presented in (5) directly into the optimization formulation. Furthermore, additional constraints related to the vehicle (wheel/road interaction, etc.), to the road profile (elevation), and to the driver will be considered.

References

- Beji, L. and Abichou, A. (2009) 'Trajectory generation and tracking of a mini-robotcraft', *Proc. of the 2009 Int. Conf. on Climbing and Walking Robots (CLAWAR09)*, Istanbul, Turkey.
- Boyd, S. and Vandenberghe, L. (2004) 'Convex Optimization', *Cambridge University Press*.
- CAMP (2004) 'Enhanced digital mapping project final report', *United States Department of Transportation*, Washington D.C., USA.
- Daniel, J., Pouly, G., Birouche, A., Lauffenburger, J-P. and Basset, M. (2009) 'Navigation-based speed profile generation for an open road speed assistant', *Proc. of the 2009 IFAC Symposium on Control in Transportation Systems (CTS09)*, Redondo Beach (CA), USA.
- Daniel, J. and Truong, C. and Lauffenburger, J-P. and Basset, M. (2009) 'Real-time Trajectory Generation for Advanced Driver Assistance Systems Applications', *Proc. of the 2009 IEEE Int. Forum On Strategic Technologies (IFOST09)*, Ho Chi Minh, Vietnam.
- De Boor, C. (1978) 'A practical guide to spline', *New-York Springer-Verlag*.
- Delingette, H. and Hebert, M. and Ikeuchi, K. (1991) 'Trajectory Generation with Curvature Constraint based on Energy Minimization', *Proc. of the 1991 Int. Workshop on Intelligent Robots and Systems (IROS91)*, Osaka, Japan.
- Dubins, L.E. (1957) 'On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents', *American Journal of Mathematics*, Vol. 79, pp.497–516.
- Eidehall, A., Pohl, J. and Gustafsson, F. (2007) 'Joint road geometry estimation and vehicle tracking', *Control Engineering Practice*, Vol. 15, pp.1484–1494.
- Floater, M.S (2008) 'On the deviation of a parametric cubic spline interpolant from its data polygon', *Computer Aided Geometric Design*, Vol. 25, pp.148–156.
- Fraichard, T. and Scheuer, A. (2004) 'From Reeds and Shepp's to Continuous-Curvature Paths', *IEEE Transaction on Robotics and Automation*, Vol. 20, pp.1025–1035.
- Gómez-Bravo, F., Cuesta, F., Ollero, A. and Viguria, A. (2008) 'Continuous curvature path generation based on β -spline curves for parking manoeuvres', *Robotic Autonomous Systems*, Vol. 56, pp.360–372.
- Graettinger, T.J. and Krogh, B. H. (1989) 'Evaluation and time-scaling of trajectories for wheeled mobile robots', *Journal of dynamic systems, measurement, and control*, Vol. 111, pp.222–231.
- Lauffenburger, J.P. and Basset, M. and Coffin, F. and Gissinger, G.L. (2003) 'Driver-aid system using path-planning for lateral vehicle control', *Control Engineering Practice*, Vol. 11, pp.217–231.
- Li, K., Tan, H.S., Misener, J. and Hedrick, K.J. (2008) 'Digital map as a virtual sensor - dynamic road curve reconstruction for a curve speed assistant', *Vehicle System Dynamics*, Vol. 46, pp.1141–1158.
- Nagy, B. and Kelly, A. (2001) 'Trajectory Generation for Car-Like Robots Using Cubic Curvature Polynomials', *Proc. of the 2001 Conference Field and Service Robots (FSR01)*, Helsinki, Finland.
- Nelson, W. (1989) 'Continuous-curvature paths for autonomous vehicles', *Proc. of the 1989 Int. Conference on Robotics and Automation (ICRA89)*, Scottsdale (AZ), USA.
- Pinchard, O. and Liegeois, A. and Pougnet, F. (1996) 'Generalized Polar Polynomials for Vehicle Path Generation with Dynamic Constraints', *Proc. of the 1996 Int. Conf. on Robotics and Automation (ICRA96)*, Minneapolis (MN), USA.

- Pollock, D.S.G. (1999) ‘Smoothing with Cubic Splines’, *Handbook of Time Series Analysis, Signal Processing, and Dynamics*, Academic Press, pp.293–322.
- Pouly, G. (2009) ‘Analysis and synthesis of advanced control laws for vehicle ground guidance’, PhD dissertation, Université de Haute-Alsace.
- Schmidt, J.W. and Heß, W. (1988) ‘Positivity of cubic polynomials on intervals and positive spline interpolation’, *BIT Numerical Mathematics*, Vol. 28, pp.340–352.
- Thomas, B., Löwenau, J. , Durekovic, S. and Otto, H.U. (2008) ‘The actmap - feedmap framework, a basis for in-vehicle adas application improvement’, *Proc. of the 2008 IEEE Intelligent Vehicles Symposium (IV’08)*, Eindhoven, Netherlands.
- Ye, J. and Qu, R. (1999) ‘Fairing of Parametric Cubic Splines’, *Mathematical And Computer Modeling*, Vol. 30, pp.121–131.